Applicant: Thomas D. FLETCHER
Serial No. 09/893,868
Response to Office Action mailed July 16, 2004

REMARKS

Claims 1, 4-17, and 19-38 are pending in this application. Claims 5, 10, 13-16, 19-25, 27-30, 32-33, and 36-38 have been amended. Claims 26 and 34 have been cancelled.

1. Reasons why the rejections under § 112 should be withdrawn

Independent claims 13 and 36 were rejected under § 112, and dependent claims 14 and 37-38 were rejected based on the independent claims.

Claim 5 has been amended to correct an error in antecedent basis. As to the feature of "Miller coupling," an article on the operation of VLSI circuits is included with this Amendment describing Miller coupling (see R. Ho, "A Primer on Noise in VLSI Systems," October, 2001, pg. 7). This article is one of countless examples of the use of the term by individuals in this art. Based on the above, Applicant submits that the phrase "Miller coupling" in claims 13 and 36 is definite and would be understood by a person of ordinary skill in this art. All recitations of the word "compliment" in the claims have been replaced with the word "complement." Reconsideration and withdrawal of the claims under 35 U.S.C. § 112, second paragraph is respectfully requested.

2. Reasons why the rejection of claims 1, 5 and 6 should be withdrawn

Claims 1, 5 and 6 were rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 3,340,388 to Earle. Looking at claim 1, this claim recites among other features, first and second clock inputs where the second clock signal is delayed from the first clock signal. Such a feature is not shown in Fig. 2 of Earle in that the same clock signal is provided to each of the carry save adders (CSAs). No delay is shown or suggested by the Earle reference. Since a feature of the claims is missing

from the Earle reference, reconsideration and withdrawal of the rejection of claims 1, 5, and 6 under 35 U.S.C. § 102(b) is respectfully requested.

3. Reasons why the rejection of claims 16, 17, 19 and 20 should be withdrawn

Claims 16, 17, 19, and 20 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,466,960 to Winters.

As amended, claim 16 recites a differential carry generate gate that has a first evaluation block and a second evaluation block that each have a plurality of transistors, "wherein each of the transistors in the first evaluation block and second evaluation block are part of a transistor stack, and wherein the number of transistors in each of said stacks is the same." It is also noted that independent claim 16 calls for the first evaluation block to be coupled to three true inputs and the second evaluation block to be coupled to three complement inputs.

In the Response to Arguments section, the Office Action contends that there are obvious errors in Fig. 3A of Winters. Applicant submits that the description at Col. 5, lines 18-55 and Fig. 3A are so replete with such alleged "errors" that it cannot be used properly for this rejection. Looking at Col. 5, lines 34-41, Winters describes the circuit of Fig. 3A to be used to perform the logic function of equation 8 (i.e., A XOR B XOR Cin; see Col.2, line 23). Given Fig. 3A and the description provided in Winters, one skilled in the art would not be led to the claimed invention. The changes that would need to be made to the description and drawing of Winters are excessive. First, the equation alluded to in the Office Action (AB+AB+BC) appears nowhere in Winters. Second, one of the "25" transistors is to be connected in a way different from that shown in the drawing. Third, one of the "26" transistors is to be connected in a way different from that shown in the drawing. Fourth, one of the gate inputs from one of the "26" transistors is to be changed. Applicant submits that the Office Action is impermissively relying on the description of the present application to allegedly correct errors in a drawing that are described to implement a completely different function.

Applicant also notes that there is no evidence of a motivation to modify the prior art references to obtain the claimed invention. *See, e.g., In re Zurko*, 258 F.3d 1379,

1368 (Fed. Cir. 2001) (holding that an Examiner must "point to some concrete evidence in the record" of a motivation to combine or modify the references to support an obviousness rejection).

For at least these reasons, claims 16 is believed to be patentable. Claims 17 and 19-20 depend from claim 16 and are patentable for at least the same reasons as claim 16, as well as for additional limitations contained therein.

4. Reasons why claims 25, 33, and 35-38 are allowable

Claims 25 and 33 have been amended to include the limitations of claims 26 and 34 respectively. Accordingly, these claims and those that depend from them should be allowed.

5. Conclusion

Applicant respectfully requests entry of the above amendments and favorable action in connection with this application. The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. 1.16 or 1.17 to Kenyon & Kenyon Deposit Account No. 11-0600. The Examiner is invited to contact the undersigned at (202) 220-4310 to discuss any matter concerning this application.

Respectfully submitted,

Kenyon & Kenyon

Date: December 7, 2005 By: _____

Shawn W. O'Dowd
Registration No. 34,687
(Attorney for Intel Corporation)

KENYON & KENYON
1500 K Street, N.W.
Washington, D.C. 20005
Ph.: (202) 220-4200
Fax.: (202) 220-4201

A primer on noise in VLSI systems

*or, Are you **sure** this damn chip works?*

Ron Ho
10/9/01
ronho@vlsi.stanford.edu

Introduction

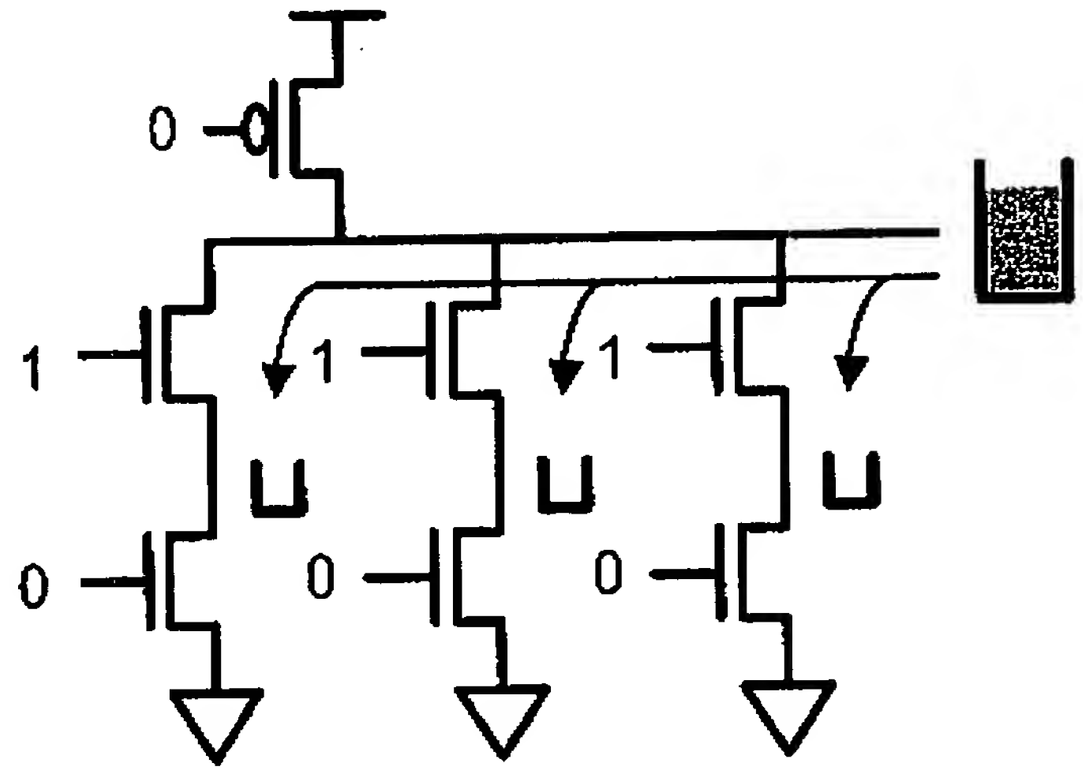
- Physical noise sources have long vexed analog designers
  - Stochastic in nature (e.g. thermal noise in resistors)
  - Why designers design "LNA"s and not simply "A"s
  - Relatively straight-forward verification
    - Except for alpha or cosmic strikes
- Deterministic noise a big & growing problem for digital designers
  - An *enfant terrible* of technology scaling
  - Exacerbated by performance-noise tradeoffs
  - Extremely complicated verification
    - So many gates! So little time!
    - Must check for functional/reliability failures *and* timing escapes

Outline

- Sources of deterministic noise
  - Transistor noise sources
  - Capacitive coupling
  - Inductive coupling
- Noise analysis
  - Templates and correct construction
  - Static noise tools
    - IBM's Harmony / CadMOS's PaciflC
- Future themes (?)

Deterministic gate noise: charge-sharing

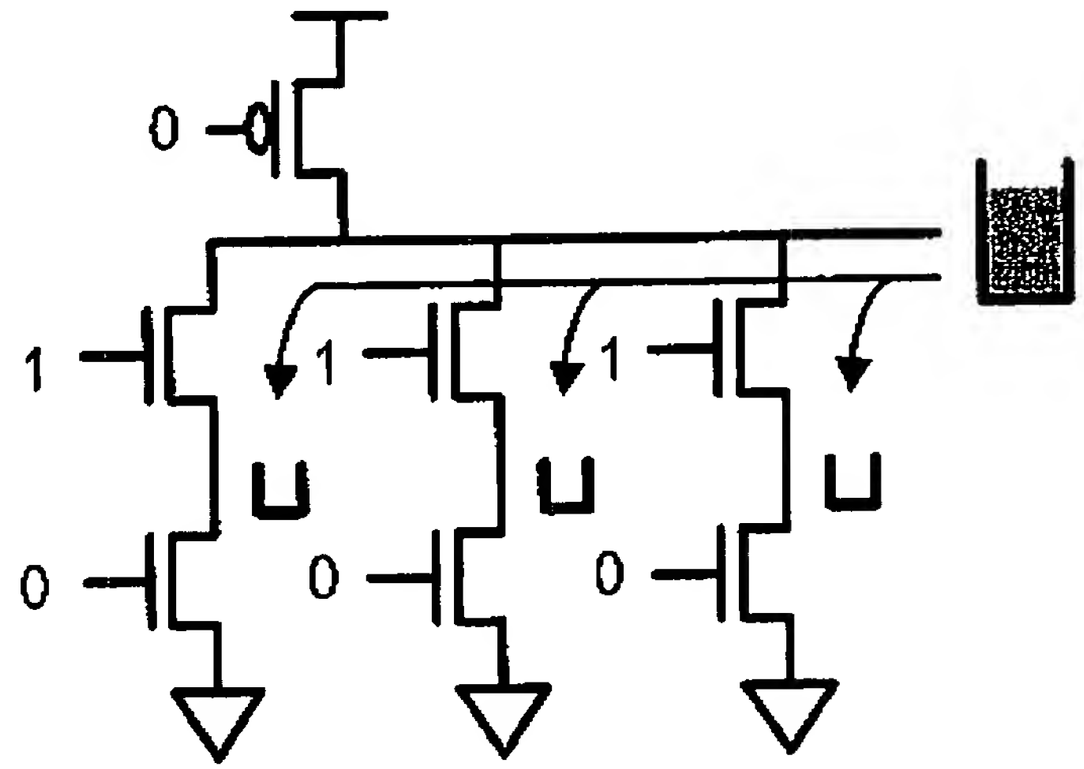
- Charge-sharing
  - Dynamic logic exemplifies performance vs. noise tradeoffs
  - Helped somewhat by feedback devices (weak pFETs)



- Not dramatically changing with technology scaling
  - Gets worse if V_t/V_{dd} falls; gets better if C_{diff} reduced (*e.g.* SOI)

Deterministic gate noise: charge-sharing

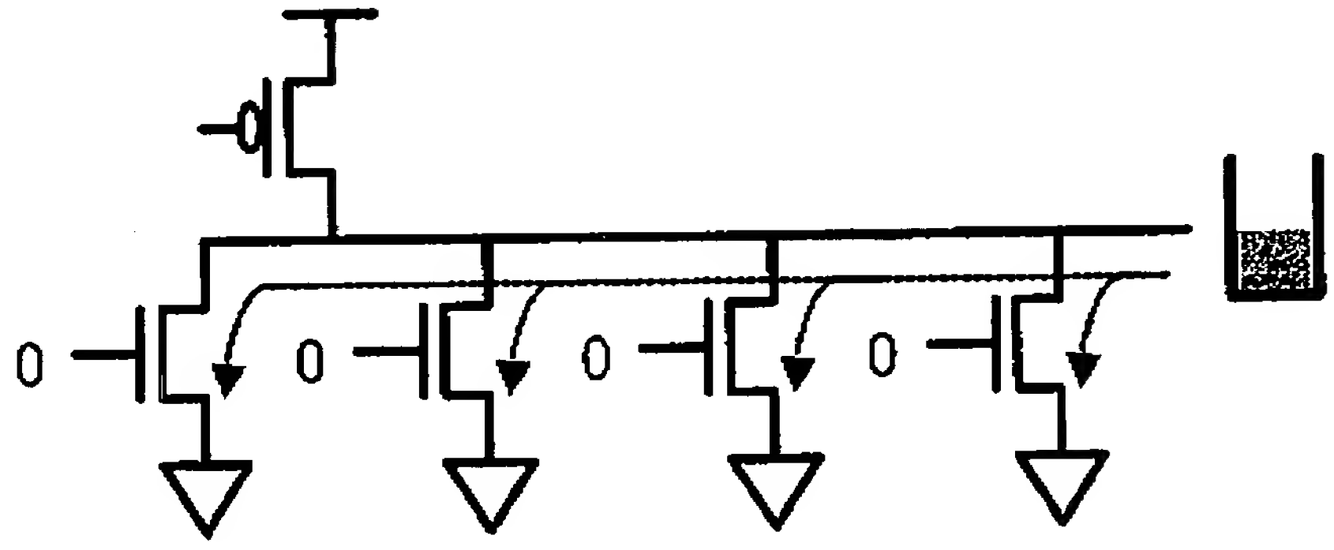
- Charge-sharing
  - Dynamic logic exemplifies performance vs. noise tradeoffs
  - Helped somewhat by feedback devices (weak pFETs)



- Not dramatically changing with technology scaling
  - Gets worse if V_t/V_{dd} falls; gets better if C_{diff} reduced (*e.g.* SOI)

Deterministic gate noise: leakage

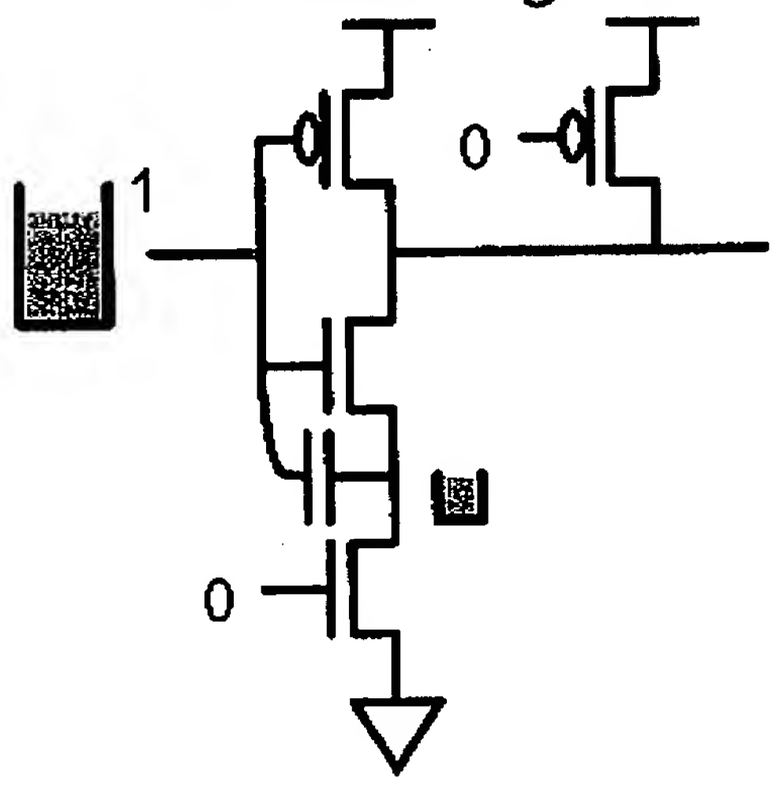
- Charge-leakage
  - Dynamic gate phenomenon
  - Helped greatly by feedback devices

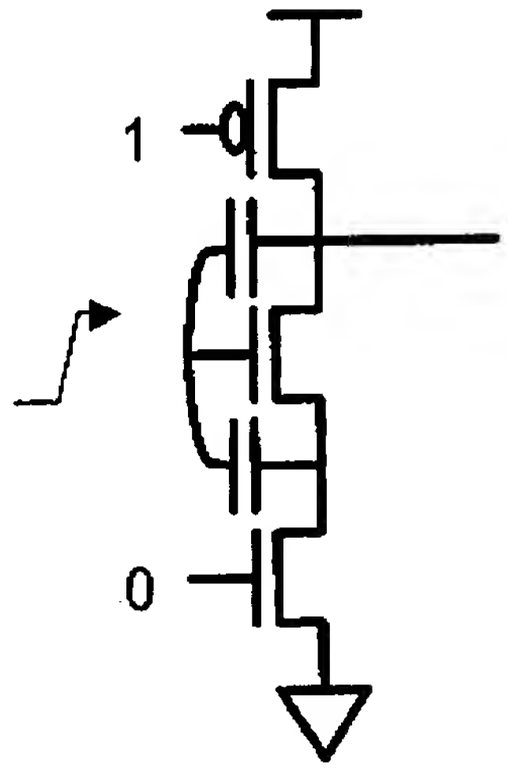


- Leakage rises exponentially as V_t falls
- Secondary effect: 10^9 devices' leakage adds up to lots of power
  - Potential for more supply noise

Deterministic gate noise: Miller coupling

- Miller coupling
  - Can be from drain to gate or vice versa
  - Can counteract charge-sharing to some extent

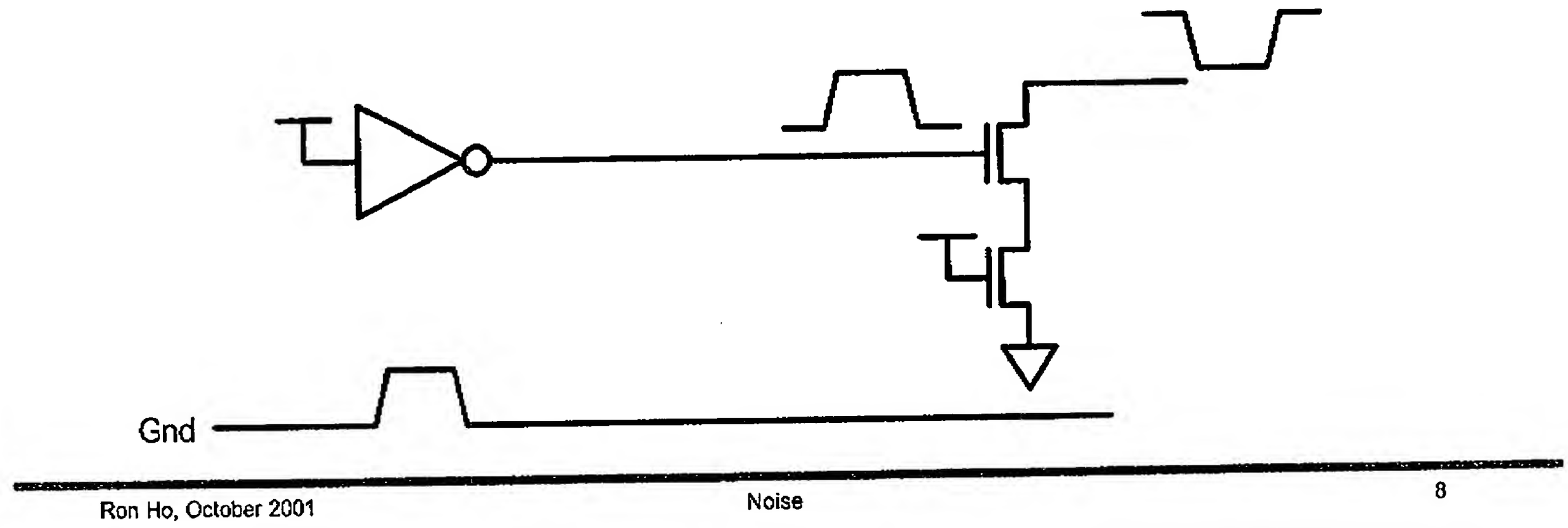




- Scales with importance of C_{gd} and C_{gs}
  - Not much change in recent technology shrinks

Deterministic gate noise: supply noise

- Supply noise has two components
  - DC IR drop from C4 bump pad
    - Gets worse farther from bump pad, eventually saturating
    - Modelled as partially collapsed power supplies
  - δI noise from simultaneous switching events (at clock edges)
    - Transients that can cause failures



Deterministic gate noise: supply noise 2

- Most high-frequency power supply current from grid, not package
  - For fast switching edges, frequency content is high
    - $f_{knee} = \frac{1}{2\pi t_{rise}}$, 30 pS t_{rise} is 5.3GHz
    - C4 package impedance increased by skin effect
  - Height of C4 balls (100μm) results in large current loop
    - C4 package impedance increased by loop inductance
    - Leads to significant power supply noise

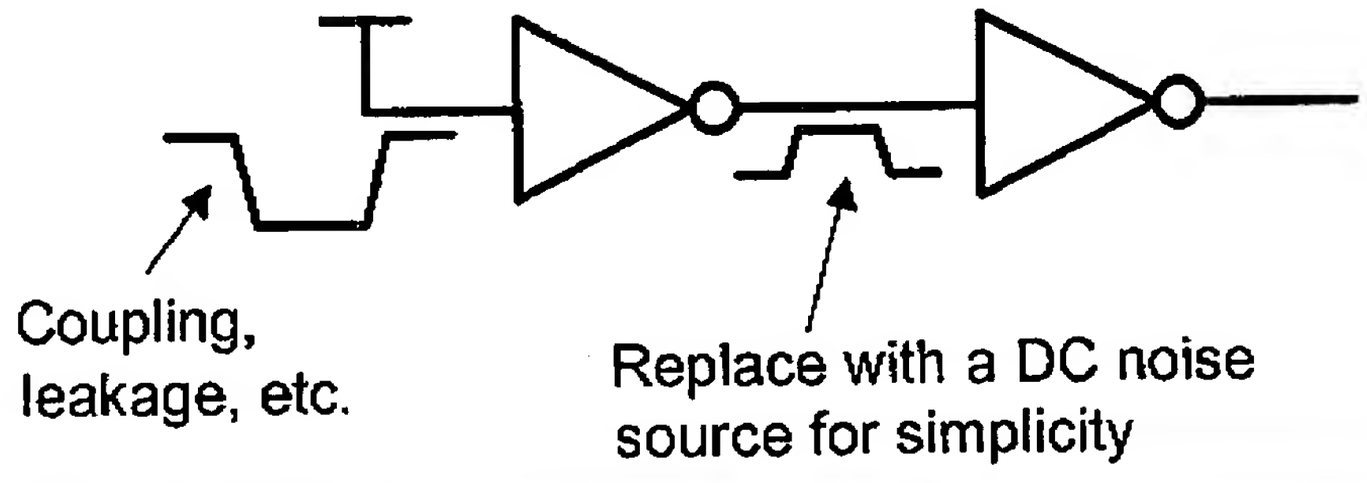
- Scaling of power supply noise in the wrong direction
  - From 0.16μm to 0.13μm, saw 15%-20% more noise
  - 2x improvement in supply impedance for constant noise

- Not well handled by noise tools beyond bump-distance mapping

Deterministic gate noise: propagated noise

- Propagated noise
  - Noise residual through a single amplifying gate
  - Affects static as well as dynamic gates

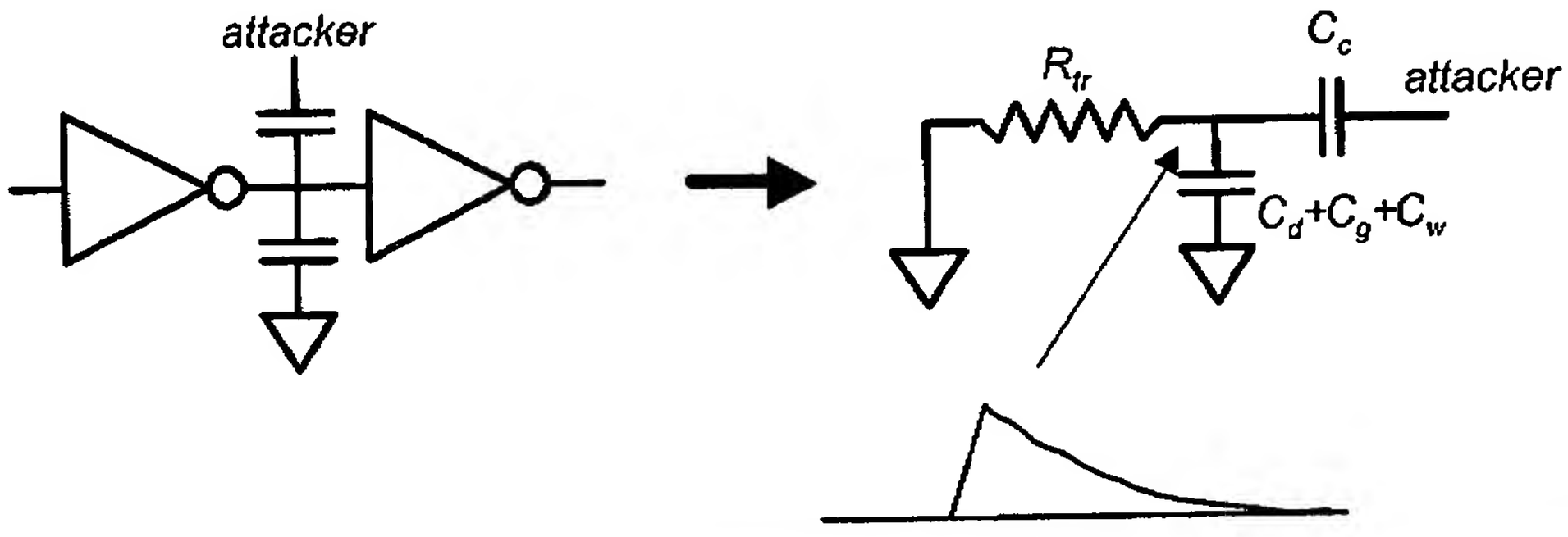
- Propagated noise is attenuated
  - More easily modeled as a DC level, even though it's not



Deterministic wire noise: capacitive coupling

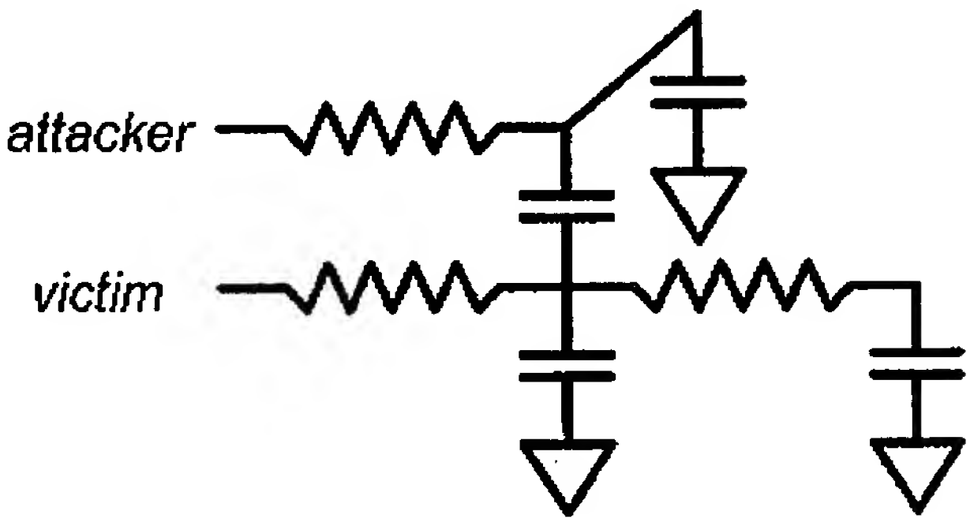
- Simplest models assume no wire resistance
  - For local (small) blocks with short wires: local interconnect
  - Attacker node is a linear ramp
  - Noise on victim is a pulse with exponential tail

$$\tau = R_{tr}(C_c + C_d + C_g + C_w)$$



Deterministic wire noise: capacitive coupling 2

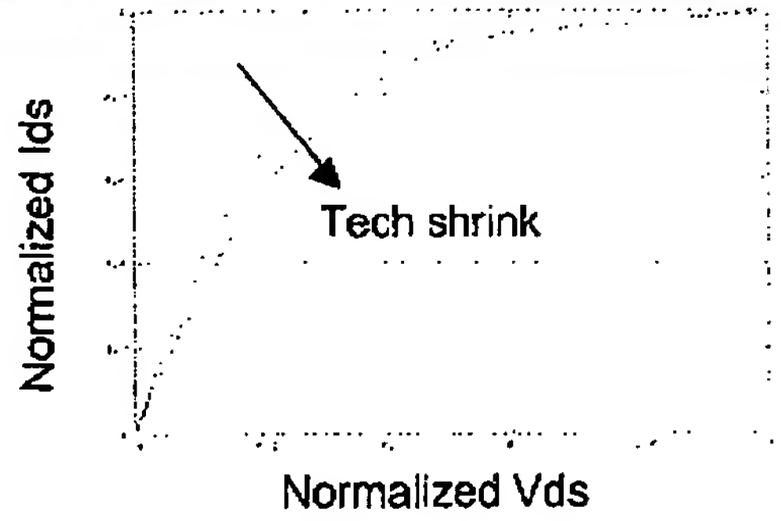
- Long wires have wire resistance
  - Attackers are often less pugnacious
  - But victims are also less able to fend off attackers



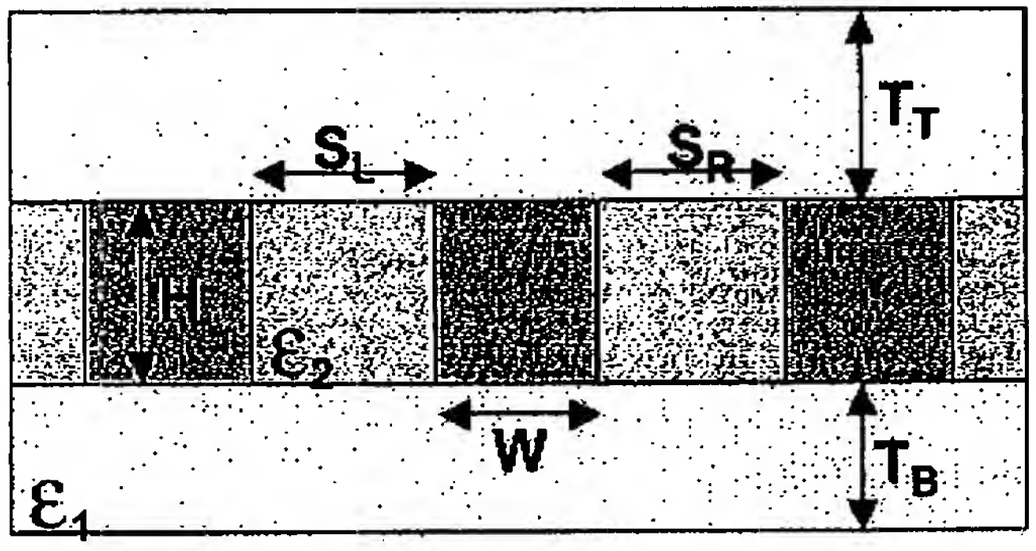
- Peak noise model can be approximated by $peak = \frac{C_c}{C_t} \frac{1}{1+\frac{\tau_{att}}{\tau_{vic}}}$
  - $\tau_{att,vic}$ are the system time constants
- Time-domain solutions typically from moment-matching models
  - Asymptotic waveform evaluation methods

Deterministic wire noise: Ccoup scaling

- Scaling of transistors
  - Switching R gets stronger
  - Linear R gets *slightly* weaker



- Scaling of wire geometries
  - Depends on metal height (H), ILD thicknesses (T), and dielectric constants (ε)



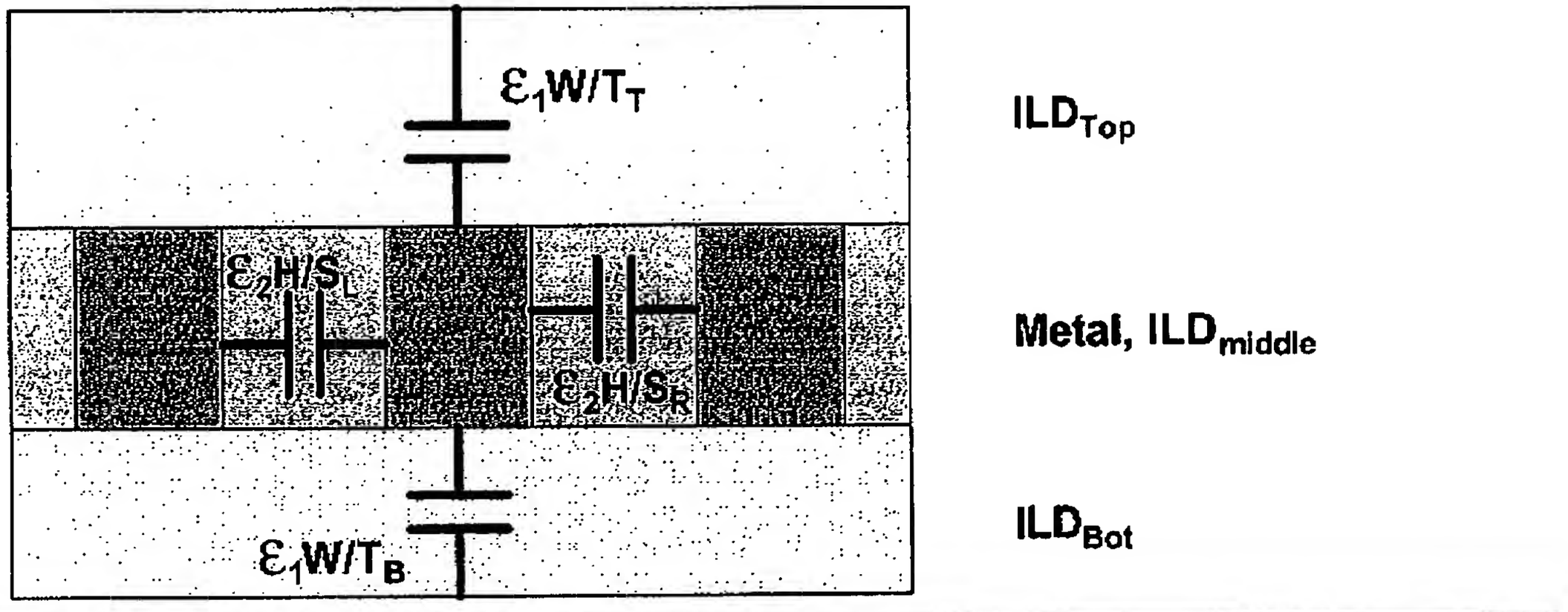
ILD_{Top}

Metal, ILD_{middle}

ILD_{Bot}

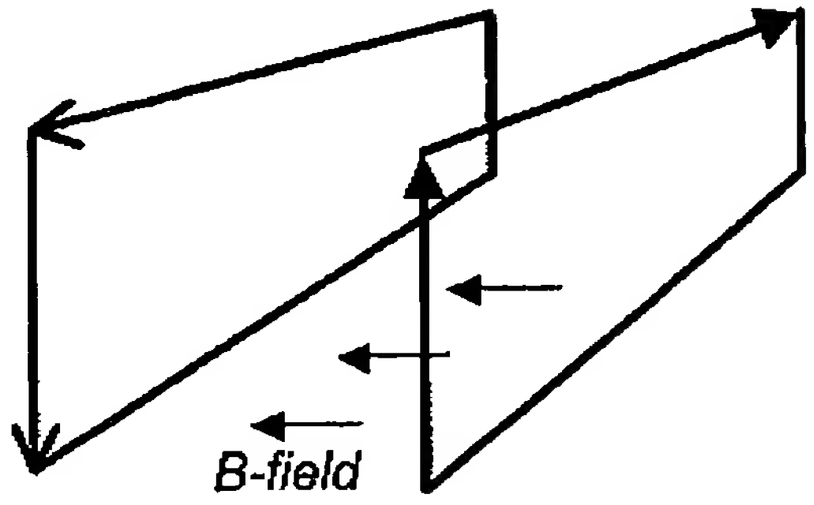
Deterministic wire noise: Ccoup scaling 2

- Capacitance per micron roughly constant
  - C = fringe (0.07fF/μm) + 4 parallel plates
  - SIA projects coupling ratios to stay constant at Cc/Ct = 70%
    - Counting on ε, or either resistance or wire density will suffer



Deterministic wire noise: inductive noise

- Wire inductance can cause two types of noise
  - Ringing from overshoot is not a big problem
    - Wires cannot ring like LC tanks because they're distributed
    - Overshoot is indicative of poor design -- catch with ERCs
  - Coupling is potentially a big problem
    - Remember Faraday and Lenz?



Induced current (falling glitch)

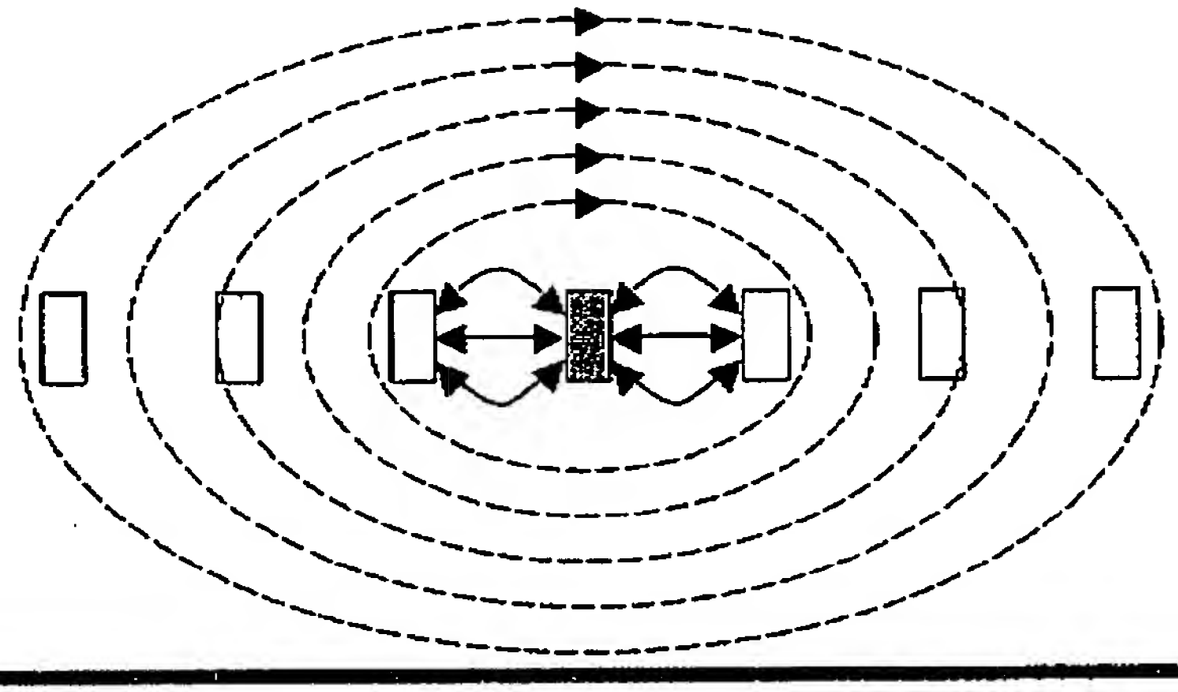
Active current (rising signal)

Oversimplification!

Here, current returns in substrate (unlikely)

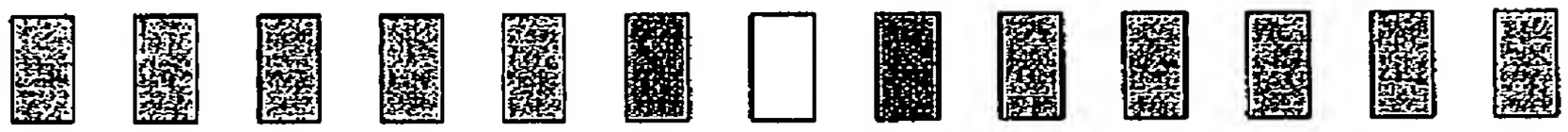
Deterministic wire noise: inductive noise 2

- Inductive noise works in opposite direction as capacitive noise
  - But it works over a much larger area
- A single wire can electrically affect only its immediate neighbors
  - It can magnetically affect all wires inside its current loop
    - Defined by the nearest current return path
    - A good way to determine how many potential attackers exist



Deterministic wire noise: inductive noise 3

- Inductive noise is important when
  - Wires are medium length (or else resistors dominate)
  - Wires are in a wide bus with many attackers
  - Not very many current return paths
    - Differential wiring helps a lot!
- Worst-case vector combines C and L noise
  - Red = rising; Grey = falling; White = victim



  - HP had this exact scenario cause a failure on a CPU
    - 0.8µm PA-RISC, 64b bus, no power lines within the bus

Deterministic wire noise: inductive noise 4

- Calculating wire inductance is computationally expensive
  - Use mathematical tricks to break chicken-egg problem
- No closed-form models of inductive noise exist
  - Rely on simulations of each situation to verify goodness
- A bigger problem is database explosion
  - Each L-extracted wire is 10x-100x the data as for C-extract
  - Cap extraction alone will need 1 TB of data today
- Scaling?
  - L or M per unit length constant
  - Both top and bottom of $\frac{\delta i}{\delta t}$ scale down; constant noise

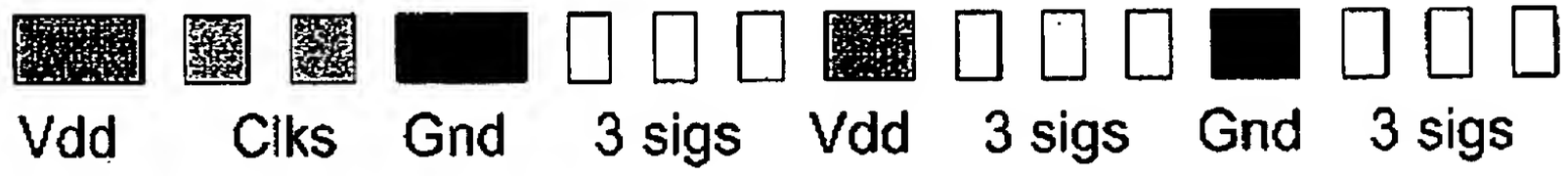
Noise analysis

- Three basic models of noise analysis
  - Template-based correct construction
    - Inexpensive analysis
    - Potentially expensive in design restrictions
  - Automated static simulation of noise
    - Expensive analysis
      - What do you do when the tool returns millions of violations?
    - Design freedom
  - Prayer

Noise analysis: template-based

- Template-based noise analysis looks at representative circuits
  - If all your circuits fit into the forms checked, you're golden
  - Your circuit is then correct by construction

- Great for inductance noise checks
  - Avoid massive and complicated global wire extraction
  - Require all wiring on global layers to fall into templates
    - e.g. M5 wiring template repeats across the chip



  - Ensure wires in this template can't fail, and stick with it
    - A much smaller simulation and extraction problem

Noise analysis: template-based 2

- Can also use for standard-cell-based design
  - Combine all cells in pairs, one driving the other
  - Simulate all worst-case noise vectors
    - *e.g.*, low-skew NOR driving domino gate
  - Model external noise sources with fixed waveforms
    - Tends to be overpessimistic
  - For small libraries and smaller designs

Noise analysis: static noise tools

- Example of Cadence/CadMOS/IBM tool
  - Called "Harmony" at IBM, "PacifIC" at CadMOS/Cadence
  - Used for noise checking on an S/390 CPU

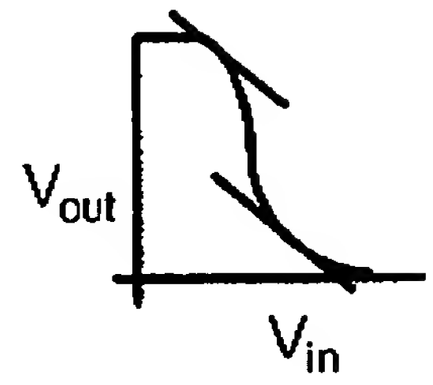
- Two tools, combined into one
  - Basic Harmony runs on macros (functional blocks)
    - An engine that simulates circuits for noise analysis
    - Assumes blocks are small enough to ignore wire resistance
    - Generates Rdriv and Cload ports at the block IOs
  - Global Harmony runs on the wires that connect the macros
    - An interconnect reduction engine for fast wire coupling sims

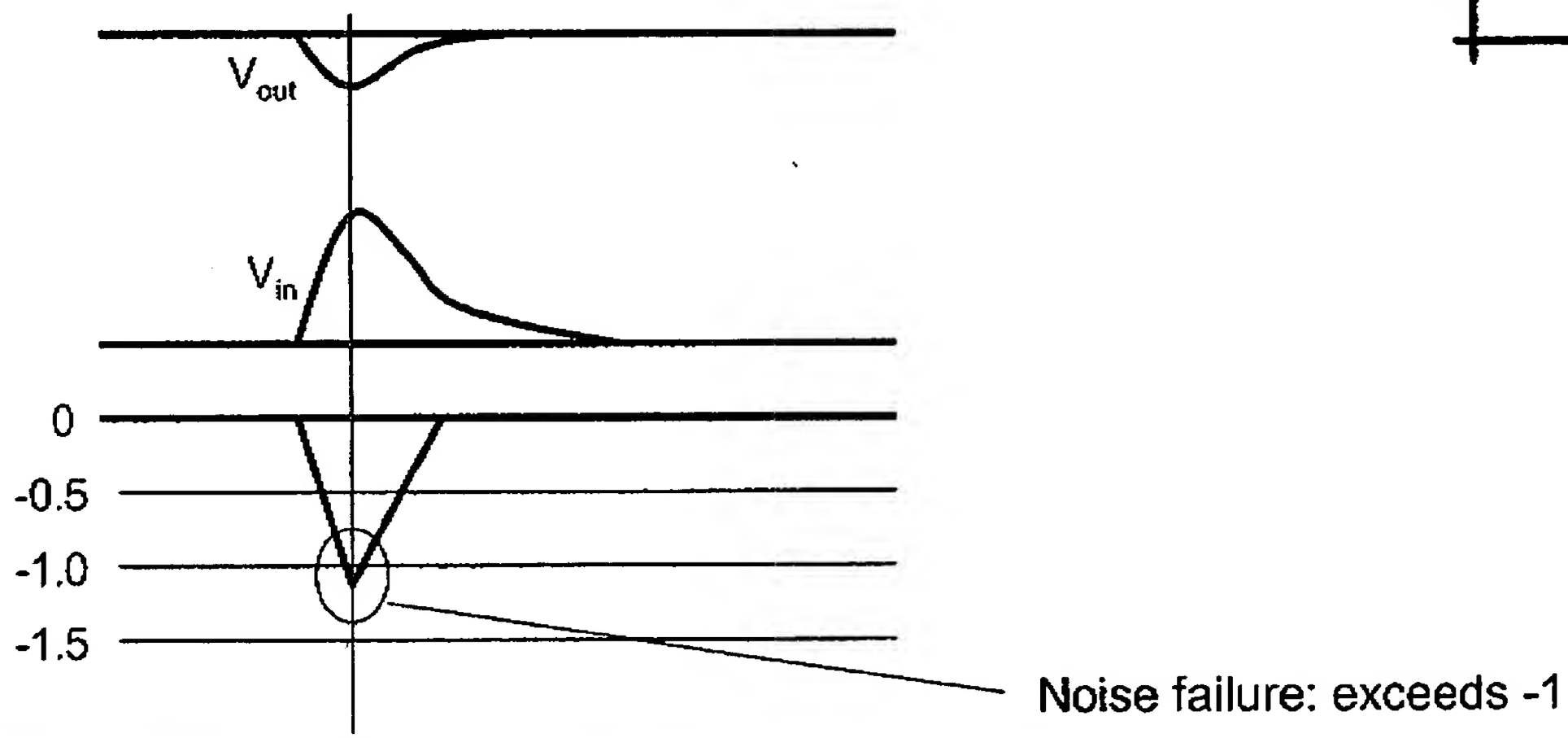
Noise analysis: Macro Harmony

- Similar in concept to static timing analysis
  - Walks a netlist, constructs directed graph
    - Breaks feedback loops for a directed acyclic graph
  - Groups channel-connected FETs together for simulation
  - Prepares a transistor-level sim for each node, noise source
    - With appropriate input stimuli
  - Takes 200 minutes for a block with 120K transistors
    - On an RS6000-590
    - Using their ACES simulation engine
    - They don't publish runtimes for Global Harmony…

Noise analysis: Macro Harmony 2

- Noise failure criteria
  - Do not use DC unity-gain noise margins $|\frac{\delta V_{out}}{\delta V_{in}}| < 1$
    - Too conservative, since gates are low-pass filters
  - Use a pulse version of this criteria





Noise analysis: Macro Harmony 3

- Doing the simulations for each and every node
  1. Establish base DC voltage levels, which may be degraded
     1. V_t drop from half-passgates
     2. Leakage current, allowed to run for a phase or so
     3. Ratioed logic
     4. Power supply noise
  2. Sensitize inputs
     1. Sensitize inputs for coupled noise (cap only; L in future?)
     2. Sensitize inputs for propagated noise (from previous node)
     3. Sensitize inputs for charge-sharing noise
  3. Find combined sensitization with largest output noise
     1. Constrain input sensitizations with hazard or mutex conds
     2. Use heuristics that understand CMOS, domino, passgate logic
  4. Calculate noise failure or success
     1. If failure, flag it; then pretend it passed and keep going

Noise analysis: Macro Harmony 4

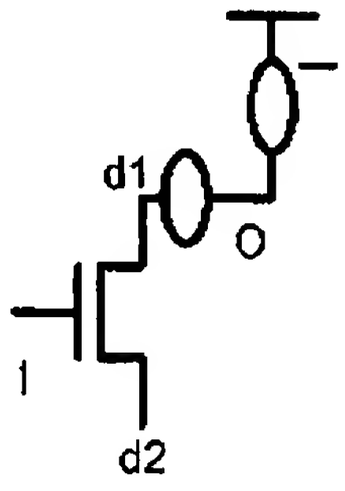
- Assumptions
  - Full-rail signalling only
  - Gates replaced by grounded capacitors to partition CCCs
  - Modelling noise sources
    - Ignore Miller coupling noise
    - Power supply noise equivalent to collapsed rails
    - Coupled noise voltages are generalized pulses
  - Superimpose all noise simulation results in time-domain
    - Linear behavior
    - Line the peaks up unless we have timing/logic reasons not to
  - At broken feedback loops, iterate
    - Assume no noise, then calculate input noise and feedback
    - Recalculate input noise and re-feedback. Lather, rinse, repeat.

Noise analysis: Macro Harmony 5

Sensitizing inputs

- Write logical constraints to figure out input vectors
  - $f_{i,j}$ = the condition for a conducting channel between *i* and *j*
  - Will a coupling attack on high net D prop to a low net O?

$$f_{O,Gnd} \cdot \overline{f_{O,D}} \cdot f_{D,Vdd} \cdot \overline{f_{D,Gnd}}$$



- Will a switching event on input I cause charge-sharing to O?
  - Let d1 and d2 be the two diffusions of nFET with gate I
  - A simplified version of the real constraint

$$[f_{O,Vdd} \cdot f_{d1,O} \cdot \overline{f_{d2,Gnd}} \cdot \overline{f_{d2,Vdd}}]_{I=0} \cdot [f_{O,Vdd}]_{I=1}$$

  - Constrain further with mutex or complementary signals

Noise analysis: Macro and Global Harmony

- When Macro Harmony reaches the end of the block
  - Models outputs with equivalent resistor pullup/dn
  - Models inputs with equivalent grounded capacitors
  - Feeds these ports into Global Harmony's net simulator

- Global Harmony is a fast coupling simulation engine
  - Takes cumbersome RC networks and reduces them
    - Creates MIMO impedance macromodels
    - Stores the macromodels in compressed binary tables
    - Uses these macromodels for global timing simulations also
  - Uses timing information to filter out attackers

Noise analysis: Global Harmony

- For each global net, identify a complex of surrounding nets
  - Secondary nets excluded if their C_c/C_t is not large enough
  - Excluded nets have their xcaps tied to ground

- For this net complex, create representative matrices
  - G (conductance), C (capacitance), and B (input/output)
  - Impedance macromodel $Z(s) = B^T(G + sC)^{-1}B$
    - Lots of matrix math. Vladimir would love this.
  - Expand around s=0, match the first few moments

- Still has the problem of modelling high-frequency behavior with a model matched around DC, but accurate enough…

Summary

- Noise is bad but manageable
  - Many different deterministic sources
  - Some are not amenable to wide-scale simulation analysis

- Noise analysis is applying techniques of static timing analysis
  - Use the same global timing engine
  - Feed back noise and timing for accurate timing convergence